



# QSM368ZP-WF&SG368Z 系列

## Linux&Ubuntu&OpenWrt

### 音频驱动开发指导

智能模块系列

版本：1.0

日期：2024-11-20

状态：受控文件

上海移远通信技术股份有限公司（以下简称“移远通信”）始终以为客户提供最及时、最全面的服务为宗旨。如需任何帮助，请随时联系我司上海总部，联系方式如下：

上海移远通信技术股份有限公司

上海市闵行区田林路 1016 号科技绿洲 3 期（B 区）5 号楼 邮编：200233

电话：+86 21 5108 6236 邮箱：[info@quectel.com](mailto:info@quectel.com)

或联系我司当地办事处，详情请登录：<http://www.quectel.com/cn/support/sales.htm>。

如需技术支持或反馈我司技术文档中的问题，请随时登录网址：

<http://www.quectel.com/cn/support/technical.htm> 或发送邮件至：[support@quectel.com](mailto:support@quectel.com)。

## 前言

移远通信提供该文档内容以支持客户的产品设计。客户须按照文档中提供的规范、参数来设计产品。同时，您理解并同意，移远通信提供的参考设计仅作为示例。您同意在设计您目标产品时使用您独立的分析、评估和判断。在使用本文档所指导的任何硬软件或服务之前，请仔细阅读本声明。您在此承认并同意，尽管移远通信采取了商业范围内的合理努力来提供尽可能好的体验，但本文档和其所涉及服务是在“可用”基础上提供给您的。移远通信可在未事先通知的情况下，自行决定随时增加、修改或重述本文档。

## 使用和披露限制

### 许可协议

除非移远通信特别授权，否则我司所提供硬软件、材料和文档的接收方须对接收的内容保密，不得将其用于除本项目的实施与开展以外的任何其他目的。

### 版权声明

移远通信产品和本协议项下的第三方产品可能包含受移远通信或第三方材料、硬软件和文档版权保护的相关资料。除非事先得到书面同意，否则您不得获取、使用、向第三方披露我司所提供的文档和信息，或对此类受版权保护的资料进行复制、转载、抄袭、出版、展示、翻译、分发、合并、修改，或创造其衍生作品。移远通信或第三方对受版权保护的资料拥有专有权，不授予或转让任何专利、版权、商标或服务商标权的许可。为避免歧义，除了正常的非独家、免版税的产品使用许可，任何形式的购买都不可被视为授予许可。对于任何违反保密义务、未经授权使用或以其他非法形式恶意使用所述文档和信息的违法侵权行为，移远通信有权追究法律责任。

### 商标

除另行规定，本文档中的任何内容均不授予在广告、宣传或其他方面使用移远通信或第三方的任何商标、商号及名称，或其缩略语，或其仿冒品的权利。

### 第三方权利

您理解本文档可能涉及一个或多个属于第三方的硬软件和文档（“第三方材料”）。您对此类第三方材料的使用应受本文档的所有限制和义务约束。

移远通信针对第三方材料不做任何明示或暗示的保证或陈述，包括但不限于任何暗示或法定的适销性或特定用途的适用性、平静受益权、系统集成、信息准确性以及与许可技术或被许可人使用许可技术相关的不侵犯任何第三方知识产权的保证。本协议中的任何内容都不构成移远通信对任何移远通信产品或任何其他硬软件、设备、工具、信息或产品的开发、增强、修改、分销、营销、销售、提供销售或以其他方式维持生产的陈述或保证。此外，移远通信免除因交易过程、使用或贸易而产生的任何和所有保证。

## 隐私声明

为实现移远通信产品功能，特定设备数据将会上传至移远通信或第三方服务器（包括运营商、芯片供应商或您指定的服务器）。移远通信严格遵守相关法律法规，仅为实现产品功能之目的或在适用法律允许的情况下保留、使用、披露或以其他方式处理相关数据。当您与第三方进行数据交互前，请自行了解其隐私保护和数据安全政策。

## 免责声明

- 1) 移远通信不承担任何因未能遵守有关操作或设计规范而造成损害的责任。
- 2) 移远通信不承担因本文档中的任何因不准确、遗漏、或使用本文档中的信息而产生的任何责任。
- 3) 移远通信尽力确保开发中功能的完整性、准确性、及时性，但不排除上述功能错误或遗漏的可能。除非另有协议规定，否则移远通信对开发中功能的使用不做任何暗示或法定的保证。在适用法律允许的最大范围内，移远通信不对任何因使用开发中功能而遭受的损害承担责任，无论此类损害是否可以预见。
- 4) 移远通信对第三方网站及第三方资源的信息、内容、广告、商业报价、产品、服务和材料的可访问性、安全性、准确性、可用性、合法性和完整性不承担任何法律责任。

版权所有 © 上海移远通信技术股份有限公司 2024，保留一切权利。

**Copyright © Quectel Wireless Solutions Co., Ltd. 2024.**

# 文档历史

## 修订记录

版本	日期	作者	变更表述
-	2023-11-28	Peter GAO	文档创建
1.0	2024-11-20	Peter GAO/ Nikola STOLIC	受控版本

## 目录

文档历史 .....	3
目录 .....	4
图片索引 .....	5
表格索引 .....	6
1 引言 .....	7
2 RK809 Codec .....	8
3 电路原理图 .....	9
4 RK809 Codec 配置流程 .....	13
4.1. 加载声卡驱动 .....	13
4.2. 配置 RK809 Codec .....	15
4.3. 配置耳机插入中断与上报节点 .....	16
4.4. 配置耳机外置 PA .....	18
4.5. 测试 Codec .....	19
5 常见问题处理 .....	22
5.1. 声卡注册失败 .....	22
5.2. 播放无声 .....	22
5.3. 录音无声 .....	22
6 附录 参考文档及术语缩写 .....	23

## 图片索引

图 1: QSM368ZP-WF 主麦克风电路原理图 .....	9
图 2: SG368Z 系列模块主麦克风电路原理图.....	9
图 3: QSM368ZP-WF 耳机电路原理图 .....	10
图 4: SG368Z 系列模块耳机电路原理图.....	10
图 5: 耳机 PA 电路原理图.....	11
图 6: QSM368ZP-WF 扬声器电路原理图.....	11
图 7: SG368Z 系列模块扬声器电路原理图 .....	12

## 表格索引

表 1：参考文档 .....	23
表 2：术语缩写 .....	23

# 1 引言

本文档主要介绍如何在移远通信 QSM368ZP-WF 和 SG368Z 系列模块中开发音频驱动。

开发过程主要包括以下几个部分：

1. 加载声卡驱动；
2. 配置 RK809 Codec；
3. 配置耳机插入中断与上报节点；
4. 配置耳机外置 PA；
5. 测试 Codec。

本文档以 RK809 Codec 为例介绍如何开发音频驱动。

## 备注

1. 本文档适用于运行 Linux 操作系统的 QSM368ZP-WF 和运行 Linux 或 Ubuntu 操作系统的 SG368Z 系列模块。
2. SG368Z 系列模块本身无法预装 OpenWrt 系统，仅提供 SDK 及开发指导供客户二次开发。若有问题，请联系移远通信技术支持。

## 2 RK809 Codec

RK809 是一个复杂的电源管理集成电路（PMIC），同时集成 Codec 功能，用于由外部电源供电的多核系统应用。

### 备注

1. RK809 集成在模块 TE-A 内部，既有 PMIC 的电源功能，也有 Codec 功能。
2. RK809 Codec 通过 I2C0 进行控制，音频数据通过 I2S1 传输。
3. RK809 Codec 只有一路 ADC，按照差分接法只能接一个麦克风，按照单端接法可以接两个麦克风使录音组成立体声。
  - QSM368ZP-WF：当前版本只有一个主麦克风，耳机没有麦克风；
  - SG368Z 系列模块：由于只有一路 ADC，两个麦克风不支持单独录音。

# 3 电路原理图

## 1. 主麦克风:

- QSM368ZP-WF:

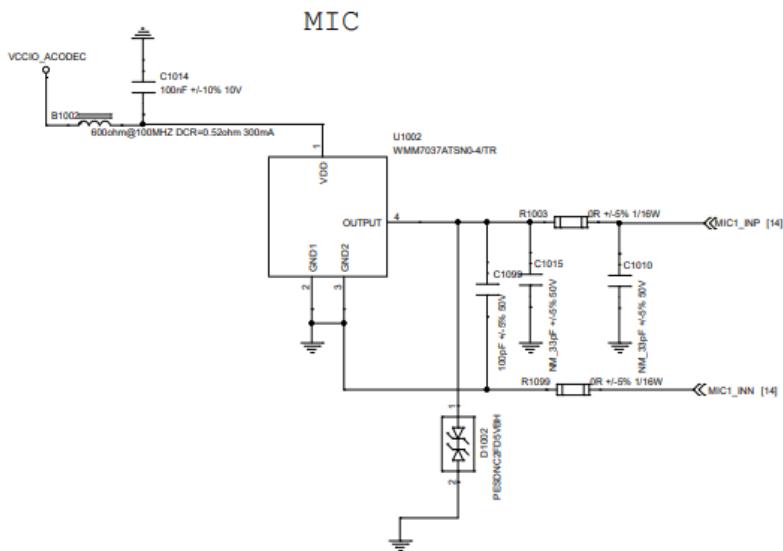


图 1: QSM368ZP-WF 主麦克风电路原理图

- SG368Z 系列模块:

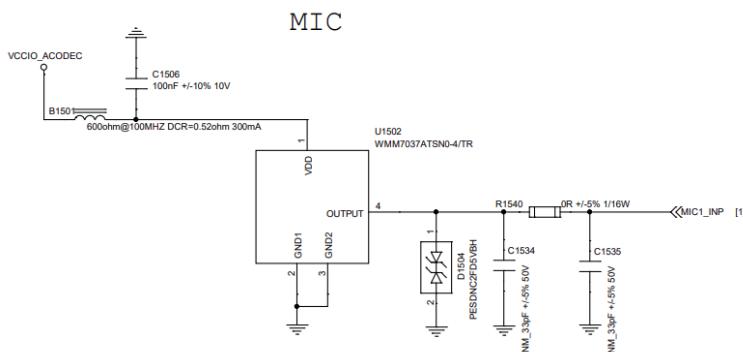


图 2: SG368Z 系列模块主麦克风电路原理图

## 2. 耳机:

- QSM368ZP-WF:

HEADSET

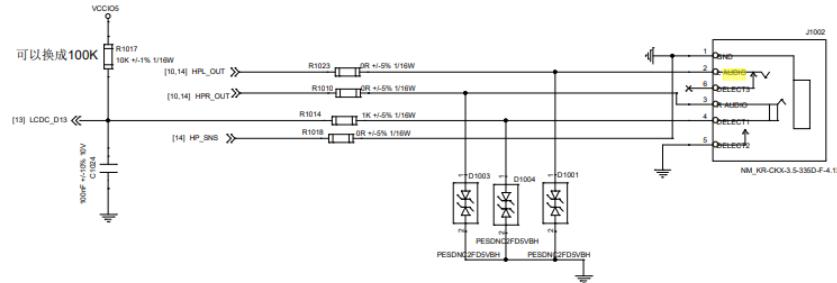


图 3: QSM368ZP-WF 耳机电路原理图

- SG368Z 系列模块:

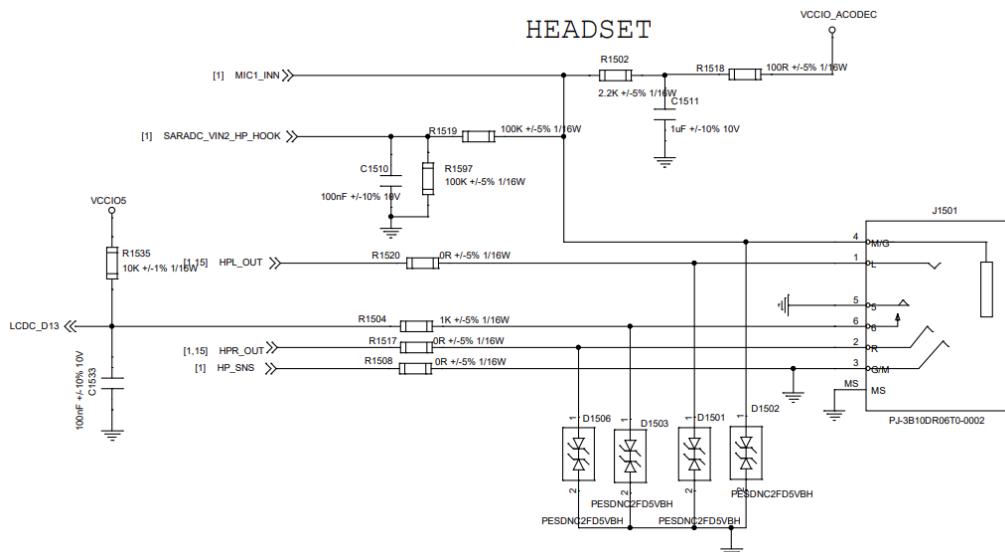


图 4: SG368Z 系列模块耳机电路原理图

### 3. 耳机 PA:

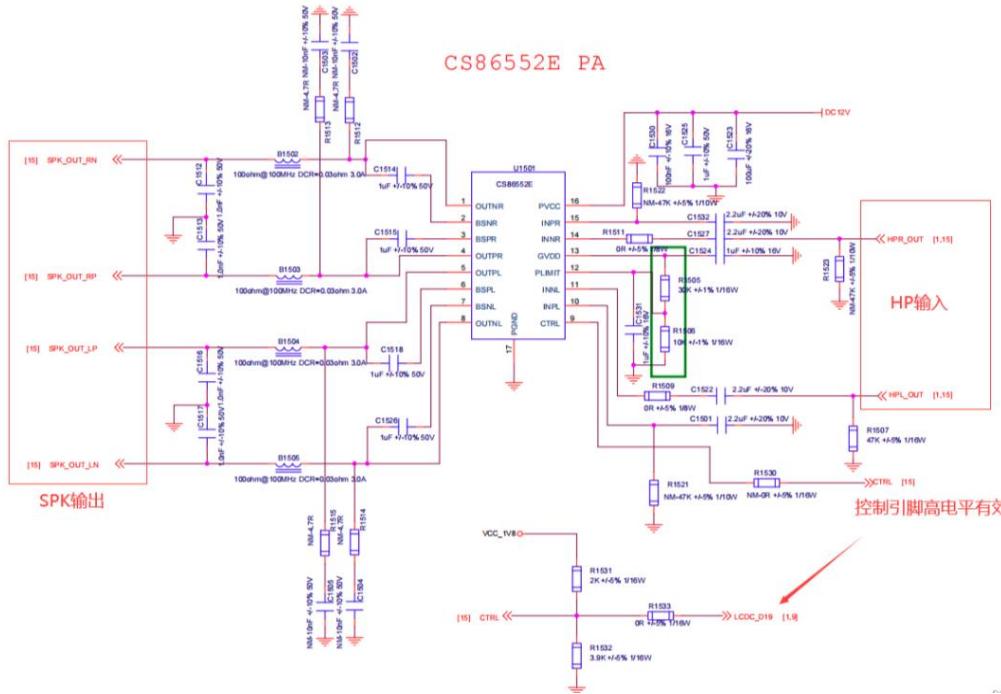


图 5: 耳机 PA 电路原理图

#### 4. 扬声器:

- QSM368Z-WF:

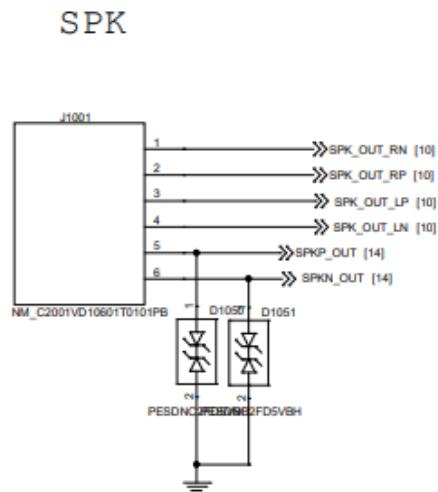


图 6: QSM368ZP-WF 扬声器电路原理图

- SG368Z 系列模块:

SPK

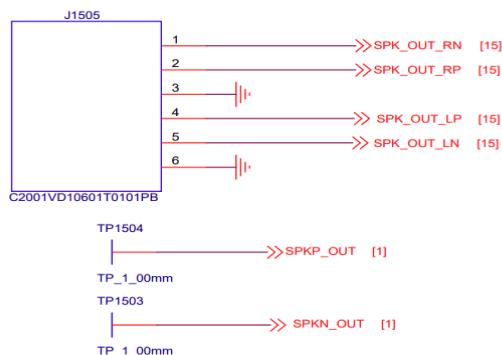


图 7: SG368Z 系列模块扬声器电路原理图

## 备注

1. RK809 Codec 的主麦克风有两种接法:
  - 接一个麦克风的差分接法（抗干扰能力更强）
  - 接两个麦克风的单端接法QSM368ZP-WF 采用差分接法，SG368Z 系列模块采用单端接法。
2. 耳机 PA 为耳机外置 PA，控制引脚为高电平时，耳机外置 PA 正常工作。

# 4 RK809 Codec 配置流程

主要驱动代码路径和设备树配置文件如下：

1. 硬件相关配置路径：

*kernel/arch/arm64/boot/dts/rockchip/rk3568-audio.dtsi*  
*kernel/arch/arm64/boot/dts/rockchip/rk3568-evb.dtsi*

2. 平台音频相关驱动路径：

*kernel/sound/soc/rockchip*

3. 主要驱动文件路径：

Machine 驱动：*kernel/sound/soc/generic/simple-card.c*  
平台驱动：*kernel/sound/soc/rockchip/rockchip\_i2s\_tdm.c*  
Codec 驱动：*kernel/sound/soc/codecs/rk817\_codec.c*  
耳机驱动：*kernel/drivers/headset\_observe/rockchip\_headset\_core.c*

## 4.1. 加载声卡驱动

1. 在设备树文件 *rk3568-audio.dtsi* 中打开声卡配置节点，进行如下配置：

- Linux 和 Ubuntu：

```
rk809_sound: rk809_sound {
    status = "okay";
    compatible = "simple-audio-card";
    simple-audio-card,format = "i2s";
    //simple-audio-card,hp-det-gpio = <&gpio3 RK_PA6 GPIO_ACTIVE_HIGH>;
    simple-audio-card,name = "rockchip_rk809-codec";
    simple-audio-card,widgets = "Headphones", "Headphones Jack";
    simple-audio-card,mclk-fs = <256>;
    simple-audio-card,cpu {
        sound-dai = <&i2s1_8ch>;
    };
    simple-audio-card,codec {
        sound-dai = <&rk809_codec>;
    };
};
```

- OpenWrt:

```
rk809_sound: rk809-sound {
    status = "okay";
    compatible = "rockchip,multicodecs-card";
    rockchip,card-name = "rockchip-rk809";
    io-channels = <&saradc 2>;
    io-channel-names = "adc-detect";
    keyup-threshold-microvolt = <595000>;
    poll-interval = <100>;
    rockchip,format = "i2s";
    rockchip,mclk-fs = <256>;
    rockchip,cpu = <&i2s1_8ch>;
    rockchip,codec = <&rk809_codec>;
    pinctrl-names = "default";
    pinctrl-0 = <&hp_det>;
    play-pause-key {
        label = "playpause";
        linux,code = <KEY_PLAYPAUSE>;
        press-threshold-microvolt = <33000>;
    };
};
```

声卡配置关键字段含义如下：

- *compatible = "simple-audio-card"*或*compatible = "rockchip,multicodecs-card"*: 声卡驱动匹配。
- *simple-audio-card,format = "i2s"*或*rockchip,format = "i2s"*: 声卡外设(Codec)挂载方式配置。
- *simple-audio-card,hp-det-gpio*: 耳机检测引脚配置。
- *simple-audio-card,cpu*或*rockchip,cpu*: 对应平台I2S1配置。
- *simple-audio-card,codec*或*rockchip,codec*: 挂载Codec配置。

2. 打开对应的I2S1节点(如下图)和Codec节点(见第4.2章)。

```
&i2s1_8ch {
    status = "okay";
    rockchip,clk-trcm = <l>;
    pinctrl-names = "default";
    pinctrl-0 = <&i2s1m0_sclktx
                &i2s1m0_lrcktx
                &i2s1m0_sdio
                &i2s1m0_sdo0>;
};
```

3. 执行 **cat /proc/asound/cards** 查看声卡是否挂载成功。若查看到挂载的声卡如下所示，则挂载成功；若挂载失败，则需检查以上配置是否正确。

- Linux 和 Ubuntu:

```
[root@RK356X:/]# cat /proc/asound/cards
0 [rockchiprk809col]: rockchip_rk809 - rockchip, rk809-codec
                           rockchip, rk809-codec
1 [rockchiphdmi]: rockchip_hdmi - rockchip, hdmi
                           rockchip, hdmi
```

- OpenWrt:

```
root@OpenWrt:/# cat /proc/asound/cards
0 [rockchiprk809]: rockchip-rk809 - rockchip-rk809
                           rockchip-rk809
```

## 4.2. 配置 RK809 Codec

配置文件路径: *kernel/arch/arm64/boot/dts/rockchip/rk3568-evb.dtsi*

1. RK809 Codec 通过 I2C0 进行通讯，需先打开 I2C0 功能。

```
&i2c0 {
    status = "okay";
```

2. 配置 RK809 Codec。

- Linux 和 Ubuntu:

```
&rk809_codec {
    #sound-dai-cells = <0>;
    compatible = "rockchip,rk809-codec", "rockchip,rk817-codec";
    clocks = <&cru I2S1_MCLKOUT>;
    clock-names = "mclk";
    assigned-clocks = <&cru I2S1_MCLKOUT>, <&cru I2S1_MCLK_TX_IOE>;
    assigned-clock-rates = <12288000>;
    assigned-clock-parents = <&cru I2S1_MCLKOUT_TX>, <&cru I2S1_MCLKOUT_TX>;
    pinctrl-names = "default";
    pinctrl-0 = <&i2s1m0_mclk>;
    hp-ctl-gpios = <&gpio3 RK_PB4 GPIO_ACTIVE_HIGH>;
    hp-volume = <20>;
    spk-volume = <3>;
    status = "okay";
};
```

- OpenWrt:

```
rk809_codec: codec {
    #sound-dai-cells = <1>;
    compatible = "rockchip,rk809-codec", "rockchip,rk817-codec";
    clocks = <&cru I2S1_MCLKOUT>;
    clock-names = "mclk";
    assigned-clocks = <&cru I2S1_MCLKOUT>, <&cru I2S1_MCLK_TX_IOE>;
    assigned-clock-rates = <12288000>;
    assigned-clock-parents = <&cru I2S1_MCLKOUT_TX>, <&cru I2S1_MCLKOUT_TX>;
    pinctrl-names = "default";
    pinctrl-0 = <&i2s1m0_mclk>;
    hp-volume = <20>;
    spk-volume = <3>;
    //mic-in-differential;
    status = "okay";
};
```

### 4.3. 配置耳机插入中断与上报节点

1. 配置耳机检测 GPIO。

**步骤 1:** 在内核版本为 4.19 的 Linux 环境中，通过 *rk\_headset* 节点检测 DET 引脚。因此，需要使能 *rk\_headset* 节点。

```
&rk_headset {
    status = "okay";
};
```

**步骤 2:** 更改 *headset\_gpio* 对应 GPIO。

- Linux 和 Ubuntu:

```
rk_headset: rk-headset {
    compatible = "rockchip_headset";
        headset_gpio = <&gpio3 RK_PA6 GPIO_ACTIVE_LOW>;
    pinctrl-names = "default";
    pinctrl-0 = <&hp_det>;
};
```

- OpenWrt:

```
&rk809_sound {
    hp-det-gpio = <&gpio3 RK_PA6 GPIO_ACTIVE_LOW>;
};
```

步骤 3: 设置 *pinctrl* 为默认下拉。

```
&pinctrl {
    headphone {
        hp_det: hp-det {
            rockchip,pins = <3 RK_PA6 RK_FUNC_GPIO &pcfg_pull_down>;
        };
    };
};
```

- 执行 **cat proc/interrupts** 查看耳机插拔是否触发中断。如下图所示，白色高亮显示区域的第 2 栏中的数字 4 表示耳机插拔触发中断次数为 4 次。

- Linux 和 Ubuntu:

118:	3	0	0	0	gpio0	13	Edge	gtx
119:	0	0	0	0	gpio0	5	Level	pericom 30216c
121:	4	0	0	0	gpio3	6	Edge	Headphone detection
122:	0	0	0	0	gpio0	23	Edge	rtw_wifi_gpio_wakeup
IPIO:	4388	8412	12502	7141	Rescheduling interrupts			

- OpenWrt:

105:	1	0	0	0	rockchip_gpio_irq	3	Level	rk817
106:	0	0	0	0	rk817	0	Edge	rk805_pwrkey_fall
107:	0	0	1	0	rk817	1	Edge	rk805_pwrkey_rise
111:	0	0	0	0	rk817	5	Edge	RTC alarm
131:	0	0	0	0	rockchip_gpio_irq	6	Edge	headset detect
IPIO:	56575	54962	60270	59360	Rescheduling interrupts			
IPII:	22380	13554	9209	6053	Function call interrupts			
IPI2:	0	0	0	0	CPU stop interrupts			

- 执行 **cat /sys/class/extcon/extcon\*/state** 查看耳机插拔状态。

```
□?2004#root@RK356X:/# cat /sys/class/extcon/extcon*/state
USB=1041
USB-HOST=0
USB_VBUS_EN=0
SDP=1
CDP=0
DCP=0
SLOW-CHARGER=0
USB=0
USB-HOST=1
USB_VBUS_EN=1
SDP=0
CDP=0
DCP=0
SLOW-CHARGER=0
MICROPHONE=0
HEADPHONE=0
                耳机拔出
                ↙
□?2004#root@RK356X:/# cat /sys/class/extcon/extcon*/state
USB=1041
USB-HOST=0
USB_VBUS_EN=0
SDP=1
CDP=0
DCP=0
SLOW-CHARGER=0
USB=0
USB-HOST=1
USB_VBUS_EN=1
SDP=0
CDP=0
DCP=0
SLOW-CHARGER=0
MICROPHONE=0
HEADPHONE=1
                耳机插入
                ↙
```

## 备注

*HEADPHONE=1* 表示耳机插入; *HEADPHONE=0* 表示耳机拔出。

## 4.4. 配置耳机外置 PA

耳机播放 HPL\_OUT 接口连接外置 PA, 音频驱动默认自带兼容(耳机或扬声器) GPIO 引脚控制的外置 PA, Linux 和 Ubuntu 系统中设备树配置如下:

```
&rk809_codec {
    #sound-dai-cells = <0>;
    compatible = "rockchip,rk809-codec", "rockchip,rk817-codec";
    clocks = <&cru I2S1_MCLKOUT>;
    clock-names = "mclk";
    assigned-clocks = <&cru I2S1_MCLKOUT>, <&cru I2S1_MCLK_TX_IOE>;
    assigned-clock-rates = <12288000>;
    assigned-clock-parents = <&cru I2S1_MCLKOUT_TX>, <&cru I2S1_MCLKOUT_TX>;
    pinctrl-names = "default";
    pinctrl-0 = <&i2s1m0_mclk &hp_det>;
    hp-ctl-gpios = <&gpio3 RK_PB4 GPIO_ACTIVE_HIGH>;
    hp-volume = <20>;
    spk-volume = <3>;
    status = "okay";
};
```

OpenWrt 系统中的 *hp-ctl-gpios* 位于 *rk3568-audio.dtsi* 文件中, 是 *pmic@20* 的一部分。

```
rk809: pmic@20 {
    rk809_codec: codec {
        hp-ctl-gpios = <&gpio3 RK_PB4 GPIO_ACTIVE_HIGH>;
    };
}.
```

由上图可知, RK809 Codec 代码中兼容 GPIO 控制的 PA。

执行 **cat /sys/kernel/debug/gpio** 查看 GPIO 信息:

```
gpiochip3: GPIOs 96-127, parent: platform/fe760000 gpio, gpio3:
  gpio-97 (          reset      ) out lo
  gpio-98 (          vcc_dvdd   ) out hi
  gpio-99 (          vcc_avdd   ) out hi
  gpio-101 (          vcc_iovdd  ) out hi
  gpio-102 ( Headphone detection ) in  lo
  gpio-108 (          hp-ctl     ) out hi
  gpio-113 (          vcc5v0_otg ) out lo
  gpio-119 (          reset      ) out hi
  gpio-121 (          vcc5v0_usb  ) out hi
  gpio-122 (          reset      ) out hi
  gpio-123 (          pwn      ) out hi
  gpio-124 (          reset      ) out hi
  gpio-125 (          pwn      ) out hi
```

由上图可知，耳机播放时，GPIO 拉高，外置 PA 生效，耳机外接的立体声扬声器正常播放。

## 4.5. 测试 Codec

移远通信提供的产品 SDK 中标配 **alsa-utils** 工具，用于手动测试音频功能。相关命令如下：

- 播放命令：**aplay**
- 录音命令：**arecord**
- 控制 Codec 内部的通路开关，设置音频流的路由和音量等：**amixer**
- 查询声卡控件，选择录音或播放路由配置：**amixer -c 0 contents**

### 备注

播放命令 **aplay** 和录音命令 **arecord** 可通过管道符连接，即 **aplay | arecord**，实现回环功能，方便驱动调试和指标测试。

示例如下：

- 查询声卡控件，选择录音或播放路由配置：

```
[?2004hroot@RK356X:/# amixer -c 0 contents
numid=2,iface=MIXER,name='Capture MIC Path'
    ; type=ENUMERATED,access=rw----,values=1,items=4
    ; Item #0 'MIC OFF'
    ; Item #1 'Main Mic'
    ; Item #2 'Hands Free Mic'
    ; Item #3 'BT Sco Mic'
    : values=1
numid=1,iface=MIXER,name='Playback Path'
    ; type=ENUMERATED,access=rw----,values=1,items=11
    ; Item #0 'OFF'
    ; Item #1 'RCV'
    ; Item #2 'SPK'
    ; Item #3 'HP'
    ; Item #4 'HP_NO_MIC'
    ; Item #5 'BT'
    ; Item #6 'SPK_HP'
    ; Item #7 'RING_SPK'
    ; Item #8 'RING_HP'
    ; Item #9 'RING_HP_NO_MIC'
    ; Item #10 'RING_SPK_HP'
    : values=3
```

```

numid=3,iface=MIXER,name='HPL Volume'
; type=INTEGER,access=rw---R--,values=1,min=0,max=255,step=0
: values=205
| dBscale-min=0.00dB,step=0.01dB,mute=1
numid=4,iface=MIXER,name='HPR Volume'
; type=INTEGER,access=rw---R--,values=1,min=0,max=255,step=0
: values=205
| dBscale-min=0.00dB,step=0.01dB,mute=1
numid=5,iface=MIXER,name='SPK Volume'
; type=INTEGER,access=rw---R--,values=2,min=0,max=255,step=0
: values=205,205
| dBscale-min=0.00dB,step=0.01dB,mute=1

```

*numid* 数值 1~5 对应的路由分别为 *Capture MIC Path*（录音通路）、*Playback Path*（播放通路）、*HPL Volume*（耳机音量左控制）、*HPR Volume*（耳机音量右控制）和 *SPK Volume*（扬声器音量控制）。

- 录音：

```
amixer -c 0 cset numid=2,iface=MIXER,name='Capture MIC Path' 1
arecord -Dhw:0,0 -c 2 -r 44100 -f s16_le /tmp/record1.wav
```

- 扬声器播放：

```
amixer -c 0 cset numid=1 2
aplay -D hw:0,0 --period-size=1024 --buffer-size=4096 -t wav /tmp/record1.wav
```

- 耳机和立体声喇叭播放：

```
amixer -c 0 cset numid=1 3
aplay -D hw:0,0 --period-size=1024 --buffer-size=4096 -t wav /tmp/record1.wav
```

- 回环（对于 QSM368ZP-WF，主麦克风录音，耳机播放；对于 SG368Z 系列模块，主麦克风和耳机麦克风录音，耳机播放）：

```
amixer -c 0 cset numid=1 3
amixer -c 0 cset numid=2 1
arecord -D hw:0,0 --period-size=1024 --buffer-size=4096 -r 48000 -c 2 -f s16_le -t raw |
aplay -D hw:0,0 --period-size=1024 --buffer-size=4096 -r 48000 -c 2 -f s16_le -t raw
```

## 备注

1. 使用 SG368Z 系列模块录音时，若未插入耳机，只能录制单声道声音（用耳机播放，一边有声音一边没有声音）。
2. amixer -c 0 cset numid= (3 4 5 音量控件) + 音量值 (0~255)。其中，255 为最大音量值。SG368Z

系列模块音量值默认为 255。

3. 扬声器和耳机音量为同一寄存器控制。设置扬声器音量后，耳机的音量会同步变化；设置耳机音量后，扬声器的音量也会同步变化。

# 5 常见问题处理

## 5.1. 声卡注册失败

1. 查看 Machine 驱动 (*kernel/sound/soc/generic/simple-card.c*) 是否执行 *probe()* 函数进行初始化。如未初始化, 请确认驱动文件是否编译至系统。
2. 使用 **kmsg** 查看内核日志定位问题原因。原因可能出现在 DAI、Codec 或者 Machine, 例如, 如下 log 表示 DAI 未注册:

```
[ 0.584114] rk-multicodecs vad-sound: ASoC: CPU DAI (null) not registered
```

- DAI 未注册: 当 DMA 通道资源不足时, 导致 DAI 注册失败, 关闭其他使用 DMA 的模块, 再次确认;
- Codec 未注册: 使用万用表和示波器测量 Codec 电压和时钟; 使用 **i2c-tools** 确认 I2C 设备通信是否正常。

## 5.2. 播放无声

- 使用 **aplay** 或者 **tinyplay** 播放频率为 1 kHz 音频文件。
- 使用示波器测量模拟输出的正弦波是否正常, 是否出现削顶尖失真的现象。
- 调整数字或者模拟增益, 观察 Codec 芯片输出端波形, 是否符合指标测试数据。
- 逐级检查引入失真的节点。

## 5.3. 录音无声

- 使用 **arecord** 或者 **tinycap** 录音, 定位问题发生在用户态还是内核态。
- 录音等待 10 秒以上确认是否为输入或输出错误。
- 使用 **amixer** 或者 **tinymix** 检查 Codec 内部 ADC 通路是否打开, 音量是否静音。
- 查看寄存器配置, 配合芯片手册或者 Codec 手册确认 IOMUX, DAI 和 Codec 配置是否正确。
- 使用万用表和示波器测量电压、时钟和数据。确认电压和时钟正常, 数据线上有波形。测量 Codec 近端模拟输入信号是否正常, 逐级定位问题点。

# 6 附录 参考文档及术语缩写

表 1: 参考文档

文档名称
[1] Quectel_SG368Z 系列_硬件设计手册
[2] Quectel_SG368Z 系列_GPIO 配置

表 2: 术语缩写

缩写	英文全称	中文全称
ADC	Analog-to-Digital Converter	模数转换器
DAI	Digital Audio Interface	数字音频接口
EVB	Evaluation Board	评估板
GPIO	General-Purpose Input/Output	通用型输入/输出
I2C	Inter-Integrated Circuit	集成电路总线
I2S	Inter-IC Sound	集成电路内置音频总线
IOMUX	Input/Output Multiplexer	输入输出多路复用器
MIC	Microphone	麦克风
PA	Power Amplifier	功率放大器
PMIC	Power Management IC	电源管理集成电路
SDK	Software Development Kit	软件开发工具包
SPK	Speaker	扬声器
TE	Terminal Equipment.	终端设备